

Modelgedreven ontwikkeling van berichtenstandaarden

StUF Expertgroep 15 november

Modelgedreven ontwikkeling

- Modelleren informatie, gegevens, berichten
- Genereren technische eindproducten

- **Kwaliteit: consistentie**
- **Scherpe en compacte schema's**

2

We willen vanuit KING StUF koppelvlakken ontwikkelen vanuit een modelgedreven aanpak. Waar we in het verleden nogal eens de standaarden maakten en beoordeelden vanuit xml-schemabestanden, willen we dat nu doen vanuit modellen. Zowel het UGM, waarin alle beschikbare entiteitstypen staan gemodelleerd, als in berichten, willen we op functioneel niveau aandacht hebben voor welke gegevens uitgewisseld moeten worden. De technische definitie, de uitwerking daarvan in xml-schema (of enig ander vorm) moet vooral op een consistente manier over alle koppelvlakken heen gebeuren, en moet voldoen aan StUF regels.

Door modelgedreven te werken, willen we bereiken:

- Dat we alle aandacht kunnen hebben op de informatiebehoefte die met berichtuitwisseling moet worden ingevuld.
- Dat de consistentie tussen verschillende modellen gewaarborgd is door controles dat van alle entiteiten en gegevens in berichten duidelijk en bekend is wat de betekenis is (uit het semantische model). En door controles dat alle entiteiten en gegevens in berichten op dezelfde manier zijn gemodelleerd conform UGM.
- Dat alle uitvoer-vormen met elkaar overeenstemmen. Dus dat berichtspecificaties in de koppelvlakspecificatie pdf exact overeenkomt met de berichtdefinitie in het xml-schema.

- Dat we scherpe en compacte schema's kunnen maken voor berichten in koppelvlakken, en die ook kunnen beheren.
- Dat we op een consistente manier functionele berichtdefinities vertalen naar technische definities (xml-schema).

SIM – UGM - KV

- **SIM:** Wat **betekenen de gegevens** die we kunnen uitwisselen, en welke **relaties** hebben die gegevens met elkaar
- **UGM:** Mogelijk maken **hergebruik** van componenten voor gegevensuitwisseling; consistent modelleren van entiteiten
- **Koppelvlak:** **Wat en hoe** moet worden gebouwd om koppelingen, in ketens en processen te realiseren

3

De modellering is gelaagd opgebouwd. We beginnen bij een semantisch informatiemodel (SIM). Hierin staan gegevens die mogelijk in een koppelvlak uitgewisseld zouden kunnen worden. Van alle objecttypen en alle eigenschappen daarvan is bekend wat de betekenis is, en hoe deze objecttypen zich tot elkaar verhouden.

Vervolgens maken we een Uitwisselingsgegevensmodel (UGM). Deze is gebaseerd op het semantische informatiemodel. Hier kunnen we het model aanpassen om deze meer geschikt te maken voor gegevensuitwisseling. Dit kennen we als “verStUffen”. Het UGM bevat entiteitstypen en elementen die mogelijk kunnen worden uitgewisseld in een koppelvlak. Het doel van het gebruik van een UGM is ervoor te zorgen dat dezelfde entiteitstypen op dezelfde manier worden gemodelleerd in alle koppelvlakken/berichten waarin dit entiteitstype voorkomt. Het UGM is dus gericht op koppelvlak-overstijgend hergebruik en consistentie.

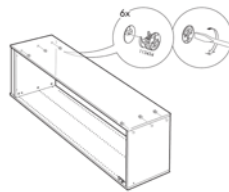
Een koppelvlak beschrijft een koppelvlak in relatie tot de ketens en processen waarin het gebruikt wordt. Dus welke referentiecomponenten daarin betrokken zijn, wanneer wat gestuurd en ingevuld moet worden, hoe dit verwerkt moet worden, enzovoort. Een onderdeel daarvan is het modelleren van de berichten die gestuurd

worden. Modelleren betekent dat bedacht en vastgelegd wordt welke gegevens we in welke berichten sturen.

SIM – UGM - KV

- SIM: "encyclopedie"
- UGM: "gereedschapskist"
- KV: "montagehandleiding"

10

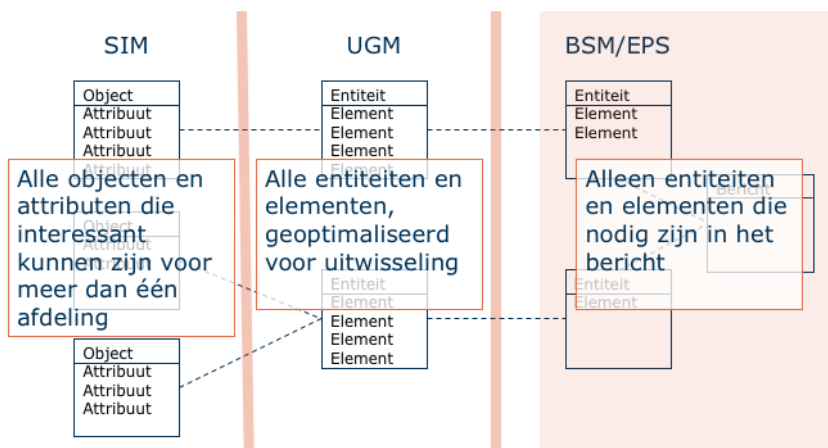


Ter illustratie zouden we het SIM kunnen zien als "encyclopedie". Hierin vinden we definities, uitleg en relaties tussen gegevens.

Het UGM kunnen we zien als "gereedschapskist". Hierin (samen met de StUF onderlaag) zitten alle onderdelen die we kunnen gebruiken bij het maken van berichten. In de gereedschapskist zitten waarschijnlijk meer dingen dan we voor een specifiek koppelvlak nodig hebben, maar andersom alles wat we nodig hebben voor een koppelvlak moet in het UGM zitten.

Het koppelvlak kun je zien als de "montagehandleiding" voor het realiseren van de services. Hierin staat welke "gereedschappen" en onderdelen we wanneer, waar en hoe gebruiken.

SIM – UGM – BSM (EPS)

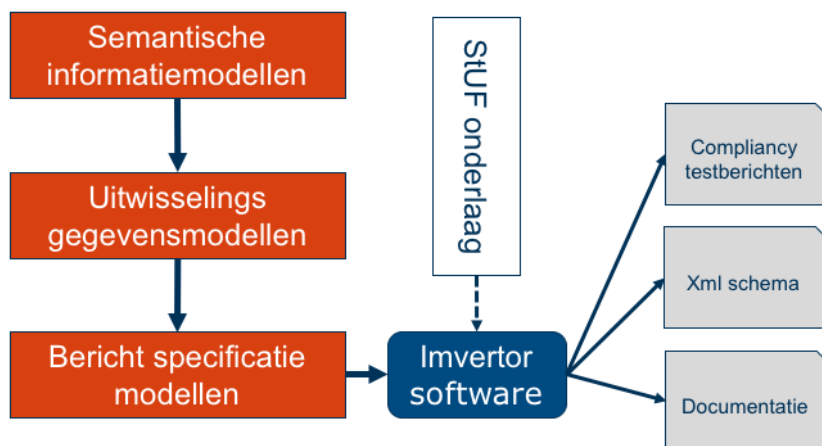


5

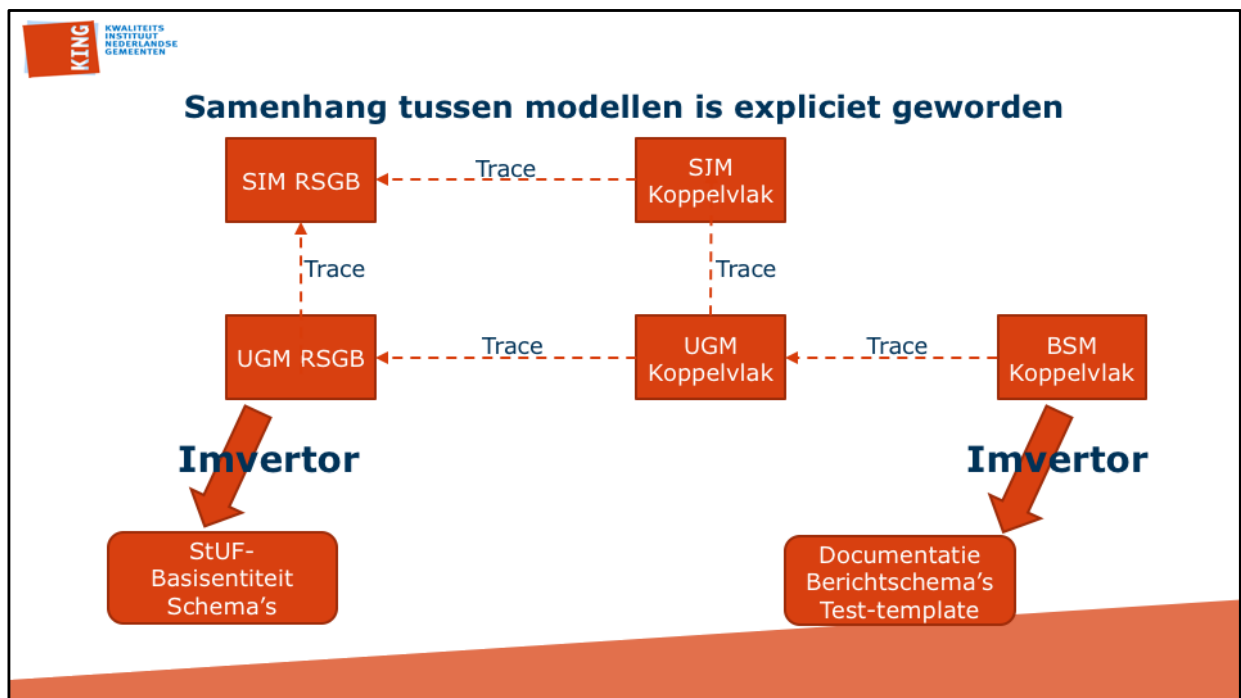
Hier zien we dat bij het verStUFfen objecten uit het SIM worden overgenomen in het UGM, maar ook anders kunnen worden gemodelleerd. Bijvoorbeeld door enkele objecttypen samen te voegen tot één entiteitstype. Ook wanneer we dat doen moet elk entiteitstype verwijzen naar een (of meerdere) objecttypen in het SIM, en moet elk element in een entiteitstype naar een attribuut in een object in het SIM.

In het berichtspecificatiemodel (BSM), dat opgesteld wordt voor/in een koppelvlak, worden entiteitstypen gebruikt uit het UGM. Vaak zal een entiteitstype niet in zijn geheel gebruikt worden, maar alleen een deel van de elementen, groepen en relaties (in xsd-termen een restriction op het UGM entiteitstype).

Modelgedreven ontwikkeling



- In Semantische modellen worden objecttypen, hun attributen en relaties gedefinieerd. Bijvoorbeeld een persoon heeft een achternaam en ouders, partner, kinderen, adres. Van elk objecttype, attribuut of relatie is beschreven wat het betekent en hoe het eruit kan zien. Bijvoorbeeld hoe lang de achternaam mag zijn, dat de postcode uit vier cijfers en 2 letters bestaat, dat geslacht man/vrouw/onbekend is.
- In uitwisselingsgegevensmodel worden keuzes en aanvullingen gedaan t.b.v. Technische gegevensuitwisseling. Sommige objecttypes worden samengevoegd. Bijvoorbeeld ingeschreven natuurlijk persoon en niet-ingeschreven natuurlijk persoon wordt samen natuurlijk persoon. Ook wordt het in woorden omschreven patroon voor postcode omgezet in door computer te interpreteren expressie (regular expression).
- In de berichtspecificatie kan gekozen worden wat in het bericht wordt opgenomen. Er kan gekozen worden elementen wel of niet op te nemen. Er kan, binnen bepaalde grenzen, ook gekozen worden eigenschappen van elementen te wijzigen, bijvoorbeeld achternaam is verplicht, of bijvoorbeeld geslacht onbekend is niet toegestaan.
- De software vertaalt de berichtspecificatie op basis van StUF regels naar verschillende technische output vormen. Bijvoorbeeld xml schema, documentatie, testberichten. Hiervoor wordt informatie gehaald uit de berichtspecificatie, uitwisselingsmodellen en informatiemodellen. Hierin kunnen gegevens uit de berichtspecificatie worden gebruikt, maar ook gegevens uit UGM (bijvoorbeeld patroon) en uit SIM (bijvoorbeeld elementdefinitie).
- In elke stap worden dus alleen dingen toegevoegd of gewijzigd wanneer afgeweken moet worden van wat er al in het bovenliggende model bekend is.
- In elke stap kan worden gekozen de standaardinvulling over te nemen, of hiervan expliciet af te wijken.
- De ontwerper van berichten kan zich puur richten op de functionele eisen en invulling. Deze hoeft zich niet bezig te houden met technische invulling of het overnemen van detailinvulling die in de SIM of UGM al bekend is.



Bij het opstellen van een koppelvlak wordt begonnen met het bepalen van de informatiebehoefte op het niveau van het semantische informatiemodel. Dit is normaal gesproken een subset van een of meerdere horizontale informatiemodellen (RSGB, RGBZ, imZTC) en/of een domeinmodel (koppelvlak SIM).

Er wordt dus gedacht en gesproken over welke SIM (bijv. RSGB) gegevens moeten worden uitgewisseld.

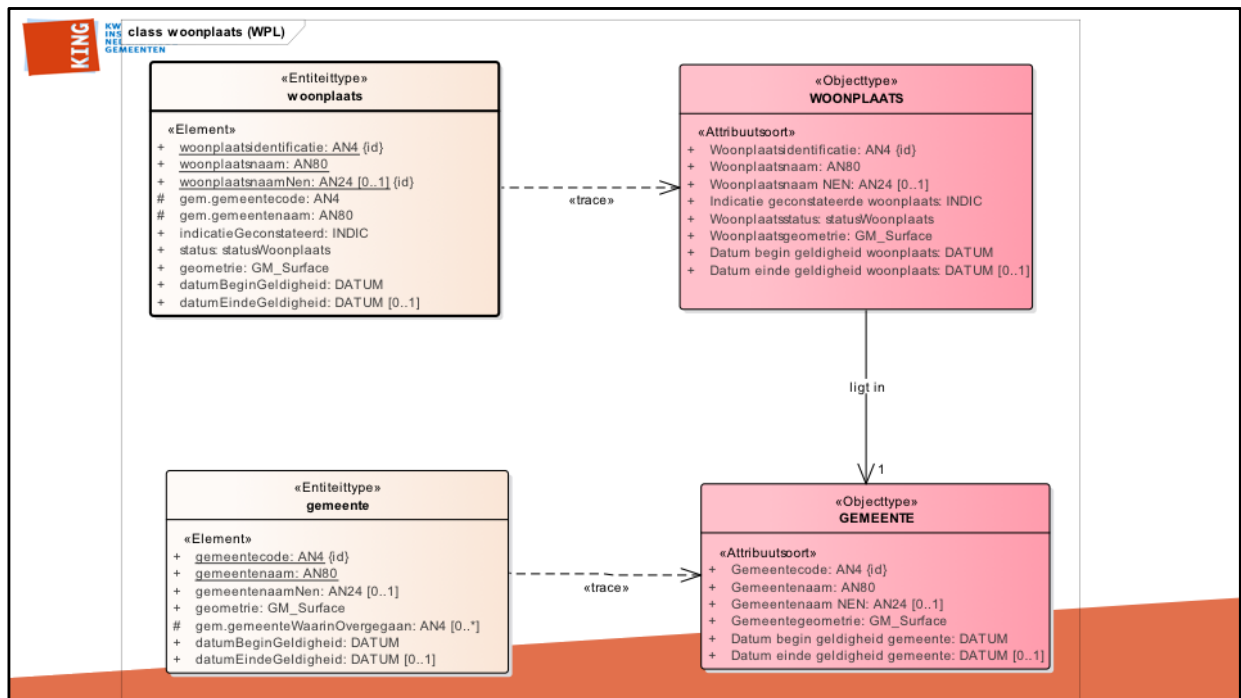
Vervolgens wordt hier een UGM van afgeleid, waarin de UGM entiteitstypen, elementen, groepen en relaties zijn opgenomen die gebruikt worden in het koppelvlak. Dit is een subset van het horizontale UGM (of soms van meerdere UGMs) en/of entiteiten afgeleid van het domeinmodel (koppelvlak SIM).

Vervolgens worden berichtspecificatiemodellen opgesteld, waarin entiteitstypen uit het koppelvlak-UGM worden gebruikt, en verbonden aan berichttypen. De software voor het genereren van berichtschema's (en andere uitvoervormen) kan op basis van de opgenomen entiteiten en de berichttypen bepalen hoe dit vertaald moet worden. Hierbij worden zo nodig extra onderdelen toegevoegd, zoals stuurgegevens, parameters, elementen als tijdvakGeldigheid en tijdstipRegistratie.

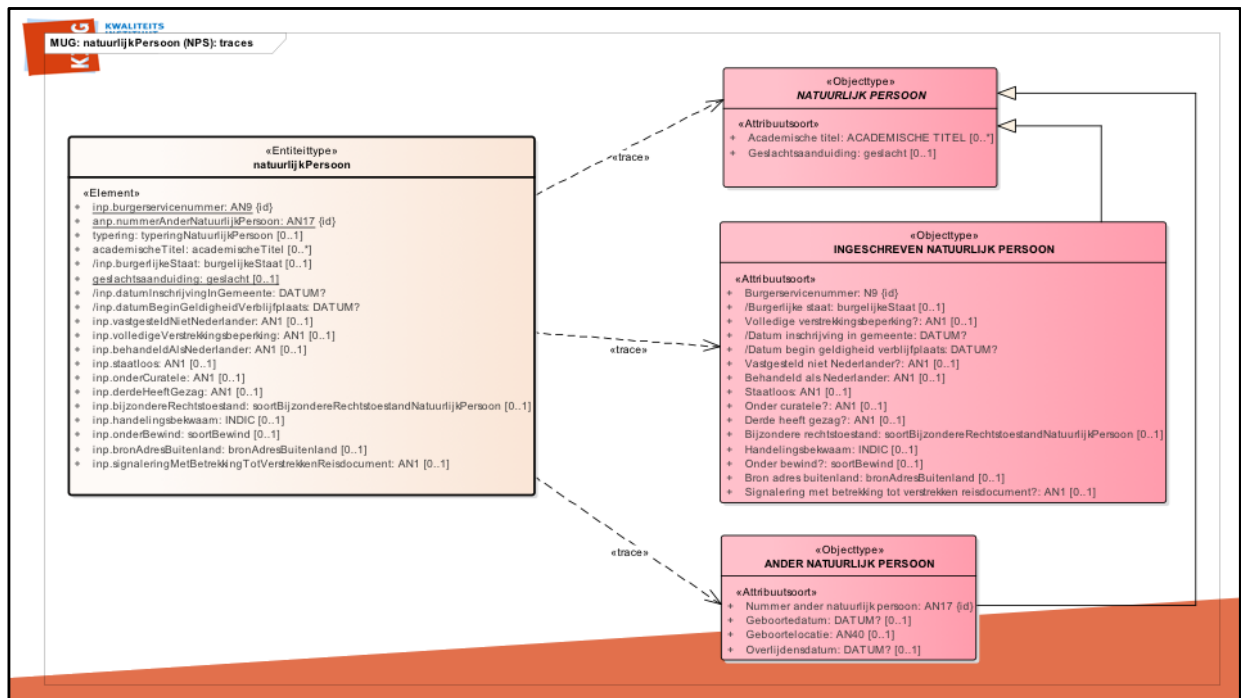
Vastleggen van de modellen

- De modellen worden opgesteld in UML
- Wij gebruiken hiervoor Enterprise Architect
- Modellen moeten ook inhoudelijk worden beoordeeld als UML

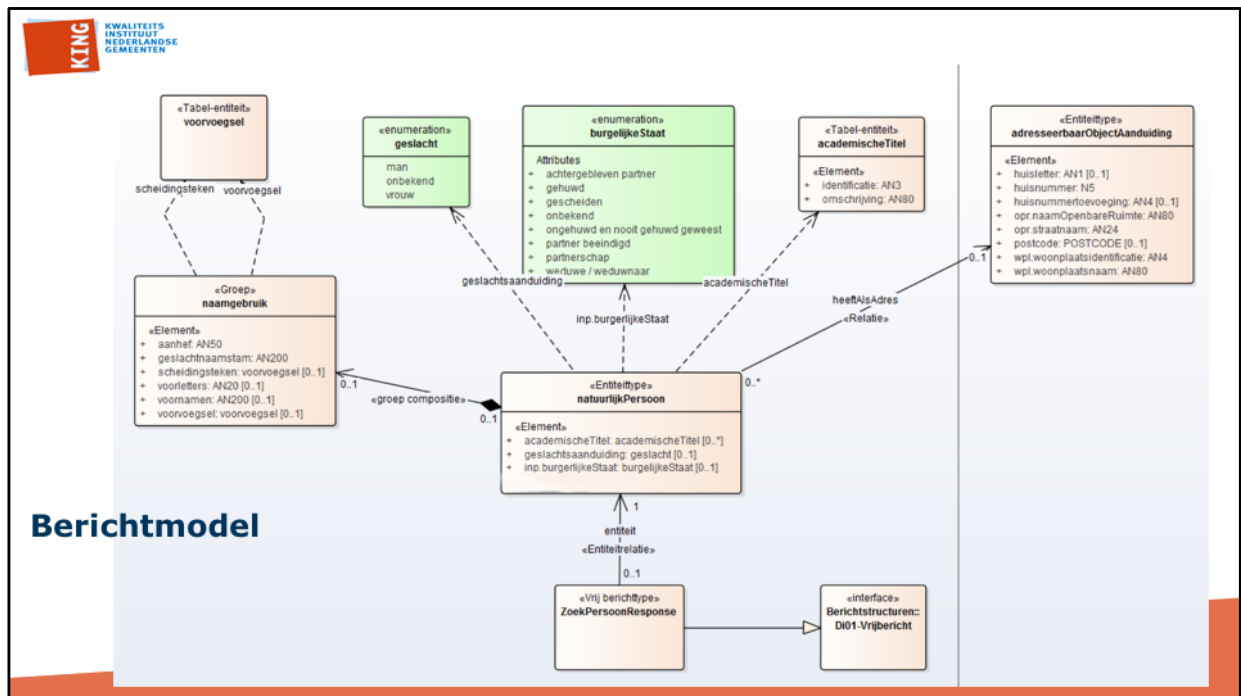
We hebben nu de gewoonte de basisentiteitenschema's te beoordelen. We zouden echter eigenlijk het UGM moeten gaan beoordelen. Daarin zien we visueel en in tekst de entiteitstypen, hun elementen, groepen en relaties. Deze informatie is ook rijker dan wat we terugzien in de basisentiteitenschema's.



Hier zien we als voorbeeld een stukje RSGB. Rechts (rood) staan de objecttypen uit het SIM. Links (geel) staan de corresponderende entiteitstypen in het UGM. Er is een trace vastgelegd tussen entiteitstypen in het UGM en de objecttypen in het SIM waar deze op gebaseerd zijn.



In dit voorbeeld zijn er in de verStUffing drie objecttypen samengevoegd tot één entiteitstype in het UGM. Toch is de “trace” naar de oorspronkelijke objecttypen nog vastgelegd en duidelijk.



Hier zien we een bericht gemodelleerd. Centraal staat entiteitstype natuurlijkPersoon. Hierin zijn niet alle gegevens van natuurlijkPersoon uit UGM BG overgenomen, slechts een deel ervan. Ook zijn (voor een deel) de relaties, groepen, enumeraties, enz. bij de natuurlijkPersoon overgenomen.

In dit voorbeeld wordt een bericht gedefinieerd, "ZoekPersoonResponse". Deze heeft als berichtstructuur Di01 en heeft een relatie "entiteit" met de natuurlijkPersoon. Dit zou vertaald worden naar een Di01 bericht met daarin een element met functie "entiteit" en entiteitstype NPS (natuurlijkPersoon).